UNIVERSITY OF BIRMINGHAM
University of Birmingham Research at Birmingham

Supervised low rank indefinite kernel approximation using minimum enclosing balls

Schleif, Frank-Michael; Gisbrecht, Andrej; Tino, Peter

DOI:
10.1016/j.neucom.2018.08.057

License:
Creative Commons: Attribution-NonCommercial-NoDerivs (CC BY-NC-ND)

Document Version
Peer reviewed version

Citation for published version (Harvard):
Schleif, F-M, Gisbrecht, A & Tino, P 2018, 'Supervised low rank indefinite kernel approximation using minimum enclosing balls', *Neurocomputing*. https://doi.org/10.1016/j.neucom.2018.08.057

Link to publication on Research at Birmingham portal

General rights
Unless a licence is specified above, all rights (including copyright and moral rights) in this document are retained by the authors and/or the copyright holders. The express permission of the copyright holder must be obtained for any use of this material other than for purposes permitted by law.

•Users may freely distribute the URL that is used to identify this publication.
•Users may download and/or print one copy of the publication from the University of Birmingham research portal for the purpose of private study or non-commercial research.
•User may use extracts from the document in line with the concept of 'fair dealing' under the Copyright, Designs and Patents Act 1988 (?)
•Users may not further distribute the material nor use it for the purposes of commercial gain.

Where a licence is displayed above, please note the terms and conditions of the licence govern your use of this document.

When citing, please reference the published version.

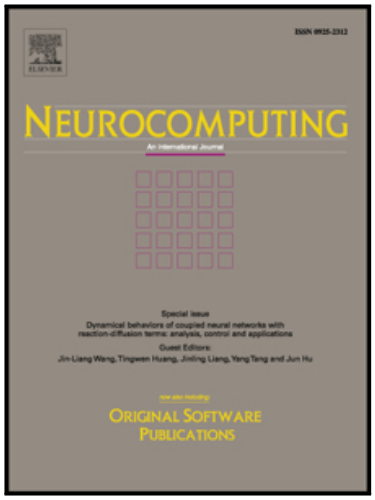
Take down policy
While the University of Birmingham exercises care and attention in making items available there are rare occasions when an item has been uploaded in error or has been deemed to be commercially or otherwise sensitive.

If you believe that this is the case for this document, please contact UBIRA@lists.bham.ac.uk providing details and we will remove access to the work immediately and investigate.

Download date: 18. Apr. 2024

Communicated by Dr. Ivor Tsang

Accepted Manuscript



Supervised low rank indefinite kernel approximation using minimum enclosing balls

Frank-Michael Schleif, Andrej Gisbrecht, Peter Tino

PII: S0925-2312(18)31019-1
DOI: https://doi.org/10.1016/j.neucom.2018.08.057
Reference: NEUCOM 19905

To appear in: *Neurocomputing*

Received date: 22 January 2018
Revised date: 28 June 2018
Accepted date: 20 August 2018

Please cite this article as: Frank-Michael Schleif, Andrej Gisbrecht, Peter Tino, Supervised low rank indefinite kernel approximation using minimum enclosing balls, *Neurocomputing* (2018), doi: https://doi.org/10.1016/j.neucom.2018.08.057

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

ACCEPTED MANUSCRIPT

Supervised low rank indefinite kernel approximation using minimum enclosing balls

Frank-Michael Schleif[1,3]*, Andrej Gisbrecht[2], Peter Tino[1]

[1] *School of Computer Science, University of Birmingham, Birmingham B15 2TT, UK*

[2] *Helsinki Institute for Information Technology, Department of Computer Science, Aalto University, Finland*

[3] *School of Computer Science, University of Appl. Sc. Wuerzburg-Schweinfurt, 97074 Wuerzburg, Germany*

Abstract

Indefinite similarity measures can be frequently found in bio-informatics by means of alignment scores, but are also common in other fields like shape measures in image retrieval. Lacking an underlying vector space, the data are given as pairwise similarities only. The few algorithms available for such data do not scale to larger datasets. Focusing on probabilistic batch classifiers, the Indefinite Kernel Fisher Discriminant (iKFD) and the Probabilistic Classification Vector Machine (PCVM) are both effective algorithms for this type of data but, with cubic complexity. Here we propose an extension of iKFD and PCVM such that linear runtime and memory complexity is achieved for low rank indefinite kernels. Employing the Nyström approximation for indefinite kernels, we also propose a new almost parameter free approach to identify the landmarks, restricted to a *supervised* learning problem. Evaluations at several larger similarity data from various domains show that the proposed methods provides similar generalization capabilities while being easier to parametrize and substantially faster for large scale data.

Keywords: indefinite kernel, kernel fisher discriminant, minimum enclosing ball, Nyström approximation, low rank approximation, classification, indefinite learning

*Corresponding author

Email addresses: frank-michael.schleif@fhws.de (Frank-Michael Schleif[1,3]), andrej.gisbrecht@aalto.fi (Andrej Gisbrecht[2]), pxt@cs.bham.ac.uk (Peter Tino[1])

ACCEPTED MANUSCRIPT

1. Introduction

Domain specific proximity measures, like alignment scores in bioinformatics [1], the modified Hausdorff-distance for structural pattern recognition [2], shape retrieval measures like the inner distance [3] and many other ones generate non-metric or indefinite similarities or dissimilarities. Classical learning algorithms like kernel machines assume Euclidean metric properties in the underlying data space and may not be applicable for this type of data.

Only few machine learning methods have been proposed for non-metric proximity data, like the indefinite kernel Fisher discriminant (iKFD) [4, 5], the probabilistic classification vector machine (PCVM) [6] or the indefinite Support Vector Machine (iSVM) in different formulations [7, 8, 9]. For the PCVM the provided kernel evaluations are considered only as basis functions and no Mercer conditions are implied. In contrast to the iKFD the PCVM is a *sparse* probabilistic kernel classifier pruning unused basis functions during training, applicable to arbitrary positive definite *and* indefinite kernel matrices. A recent review about learning with indefinite proximities can be found in [10].

While being very efficient these methods do not scale to larger datasets with in general cubic complexity. In [11, 12] the authors proposed a few Nyström based (see e.g. [13]) approximation techniques to improve the scalability of the PCVM for low rank matrices. The suggested techniques use the Nyström approximation in a non-trivial way to provide *exact* eigenvalue estimations also for *indefinite* kernel matrices. This approach is very generic and can be applied in different algorithms. In this contribution we further extend our previous work and not only derive a low rank approximation of the indefinite kernel Fisher discriminant, but also address the landmark selection from a novel view point. The obtained Ny-iKFD approach is linear in runtime and memory consumption, for low rank matrices. The formulation is exact if the rank of the matrix equals the number of independent landmarks points. The selection of the landmarks of the Nyström approximation is a critical point addressed in previous work (see e.g. [14, 15, 16]). Most recently leverage scores [17] have been found very promising, but with quadratic costs. In general these strategies use the full positive semi-definite

ACCEPTED MANUSCRIPT

(psd) kernel matrix or expect that the kernel is of some standard class like an RBF kernel. In each case the approaches presented so far are costly in runtime and memory consumption as can be seen in the subsequent experiments.

Additionally, former approaches for landmark selection aim on generic matrix reconstructions of positive semi definite (psd) kernels. We propose a restricted reconstruction of the psd or non-psd kernel matrix with respect to a *supervised* learning scenario only. We no longer expect to obtain an accurate kernel reconstruction from the approximated matrix (e.g. by using the Frobenius norm) but are pleased if the approximated matrix preserves the class boundaries in the data space.

In [12] the authors derived methods to approximate large proximity matrices by means of the Nyström approximation and conversion rules between similarities and dissimilarities. These techniques have been applied in [11] and [18] in a proof of concept setting, to obtain approximate models for the Probabilistic Classification Vector Machine and the Indefinite Fisher Kernel Discriminant analysis using a random landmark selection scheme. This work is substantially extended and detailed in this article with a specific focus on *indefinite* kernels, only. A novel landmark selection scheme is proposed. Based on this new landmark selection scheme we provide detailed new experimental results and compare to alternative landmark selection approaches. The paper provides the following improvements over the current state of the art: (1) A linear costs approximation scheme for the Indefinite Kernel Fisher Discriminant (iKFD) and the probabilistic classification vector machine (PCVM) is provided. (2) A new supervised landmark selection scheme is proposed which can be also applied to indefinite input kernels to obtain a Nystroem approximation of the given indefinite kernel. (3) A variety of experimental results is provided showing the efficiency of the proposed approach and linked to related work.

Structure of the paper: First we give some basic notations necessary in the subsequent derivations. Then we review iKFD and PCVM as well as some approximation concepts proposed by the authors in [11] which are based on the well known Nyström approximation. Subsequently, we consider the landmark selection problem in more detail and show empirically results motivating a supervised selection strategy. Finally we detail the reformulation of iKFD and PCVM based on the introduced concepts and

ACCEPTED MANUSCRIPT

show the efficiency in comparison to Ny-PCVM and Ny-iKFD for various indefinite proximity benchmark data sets.

2. Methods

2.1. Notation and basic concepts

Consider a collection of N objects $\mathbf{x}_i$, $i = 1, 2, ..., N$, in some input space $\mathcal{X}$. Given a similarity function or inner product on $\mathcal{X}$, corresponding to a metric, one can construct a proper Mercer kernel acting on pairs of points from $\mathcal{X}$. For example, if $\mathcal{X}$ is a finite dimensional vector space, a classical similarity function is the Euclidean inner product (corresponding to the Euclidean distance) - a core component of various kernel functions such as the famous radial basis function (RBF) kernel. Now, let $\phi : \mathcal{X} \mapsto \mathcal{H}$ be a mapping of patterns from $\mathcal{X}$ to a Hilbert space $\mathcal{H}$ equipped with the inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$. The transformation ϕ is in general a non-linear mapping to a high-dimensional space $\mathcal{H}$ and may in general not be given in an explicit form. Instead, a kernel function $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is given which encodes the inner product in $\mathcal{H}$. The kernel k is a positive (semi) definite function such that $k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle_{\mathcal{H}}$, for any $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$. The matrix $K_{i,j} := k(\mathbf{x}_i, \mathbf{x}_j)$ is an $N \times N$ kernel (Gram) matrix derived from the training data. The motivation for such an embedding comes with the hope that the non-linear transformation of input data into higher dimensional $\mathcal{H}$ allows for using linear techniques in $\mathcal{H}$. Kernelized methods process the embedded data points in a feature space utilizing only the inner products $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ (kernel trick) [19], without the need to explicitly calculate ϕ. The kernel function can be very generic. Most prominent are the linear kernel with $k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$ where $\langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$ is the Euclidean inner product and ϕ identity mapping, or the RBF kernel $k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{||\mathbf{x}-\mathbf{x}'||^2}{2\sigma^2}\right)$, with $\sigma > 0$ as a free scale parameter. In any case, it is always assumed that the kernel function $k(\mathbf{x}, \mathbf{x}')$ is positive semi definite (psd). This assumption is however not always fulfilled, and the underlying similarity measure may not be metric and hence not lead to a Mercer kernel. Examples can be easily found in domain specific similarity measures as mentioned before and detailed later on. Such similarity measures imply *indefinite*

ACCEPTED MANUSCRIPT

kernels, preventing standard "kernel-trick" methods developed for Mercer kernels to be applied.

For a matrix A, A^{-1} denotes the inverse of A. We will still use this notation even when A is non-regular. In that case A^{-1} will represent an inverse obtained through an Singular Value Decomposition (SVD) - based regularization.

In what follows we will review some basic concepts and approaches related to such non-metric situations.

2.2. Krein and Pseudo-Euclidean spaces

A Krein space is an *indefinite* inner product space endowed with a Hilbertian topology.

Definition 1 (Inner products and inner product space) *Let $\mathcal{Q}$ be a real vector space. An inner product space with an indefinite inner product $\langle \cdot, \cdot \rangle_{\mathcal{Q}}$ on $\mathcal{Q}$ is a bi-linear form where all $f, g, h \in \mathcal{Q}$ and $\alpha \in \mathbb{R}$ obey the following conditions.*

- *Symmetry:* $\langle f, g \rangle_{\mathcal{Q}} = \langle g, f \rangle_{\mathcal{Q}}$
- *linearity:* $\langle \alpha f + g, h \rangle_{\mathcal{Q}} = \alpha \langle f, h \rangle_{\mathcal{Q}} + \langle g, h \rangle_{\mathcal{Q}}$*;*
- $\langle f, g \rangle_{\mathcal{Q}} = 0 \; \forall g \in \mathcal{Q}$ *implies* $f = 0$

An inner product is positive definite if $\forall f \in \mathcal{Q}$, $\langle f, f \rangle_{\mathcal{Q}} \geq 0$, negative definite if $\forall f \in \mathcal{Q}$, $\langle f, f \rangle_{\mathcal{Q}} \leq 0$, otherwise it is indefinite. A vector space $\mathcal{Q}$ with inner product $\langle \cdot, \cdot \rangle_{\mathcal{Q}}$ is called an inner product space.

Definition 2 (Krein space and pseudo-Euclidean space) *An inner product space $(\mathcal{Q}, \langle \cdot, \cdot \rangle_{\mathcal{Q}})$ is a Krein space if we have two Hilbert spaces $\mathcal{H}_+$ and $\mathcal{H}_-$ spanning $\mathcal{Q}$ such that $\forall f \in \mathcal{Q}$ we have $f = f_+ + f_-$ with $f_+ \in \mathcal{H}_+$ and $f_- \in \mathcal{H}_-$ and $\forall f, g \in \mathcal{Q}$, $\langle f, g \rangle_{\mathcal{Q}} = \langle f_+, g_+ \rangle_{\mathcal{H}_+} - \langle f_-, g_- \rangle_{\mathcal{H}_-}$. A finite-dimensional Krein-space is a so called pseudo-Euclidean space (pE).*

Indefinite kernels are typically found through domain specific non-metric similarity functions (such as alignment functions used in biology [1]), specific kernel functions

ACCEPTED MANUSCRIPT

(e.g. the Manhattan kernel $k(\mathbf{x}, \mathbf{x}') = -||\mathbf{x} - \mathbf{x}'||_1$, tangent distance kernel [20]), or divergence measures plugged into standard kernel functions [21]. Another source of non-psd kernels are noise artifacts on standard kernel functions [7].

In such spaces vectors can have negative squared "norm", negative squared "distances" and the concept of orthogonality is different from the usual Euclidean case. In the subsequent experiments our input data are in general given by a symmetric indefinite kernel matrix K. We will use the symbol K to denote kernel matrices, whether psd or not. It will be clear from the context if the underlying space is a Hilbert or a Krein space. We use the symbol $\mathbf{S}$ for (symmetric) similarity matrices and $\mathbf{D}$ for a symmetric dissimilarity matrix.

In practical applications it may also happen that the given data are represented by non-metric dissimilarities. A prominent example is the dynamic timewarping score matrix which can be considered as a dissimilarity matrix of pairwise sequence alignments. Given a symmetric *dissimilarity* matrix $\mathbf{D}$ with zero diagonal [1], an embedding of the data in a pseudo-Euclidean vector space determined by the eigenvector decomposition of the associated similarity matrix[2] $\mathbf{S}$ is always possible [23].

Given the eigendecomposition of $\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$, we can compute the corresponding vectorial representation $\mathbf{V}$ of the data in the pseudo-Euclidean space by

$$\mathbf{V} = \mathbf{U}_{p+q+z} \left|\mathbf{\Lambda}_{p+q+z}\right|^{1/2}, \tag{1}$$

where $\mathbf{\Lambda}_{p+q+z}$ is a diagonal matrix containing p positive, q negative and z zero eigenvalues of $\mathbf{S}$. $\mathbf{U}_{p+q+z}$ consists of the corresponding eigenvectors. The triplet (p, q, z) is also referred to as the signature of the Pseudo-Euclidean space. This operation is however very costly and should be avoided for larger data sets. A detailed presentation of similarity and dissimilarity measures, and mathematical aspects of metric and non-metric spaces is provided in [22].

[1] A similarity matrix can be easily converted into squared dissimilarities using $d^2(\mathbf{x}, \mathbf{y}) = k(\mathbf{x}, \mathbf{x}) + k(\mathbf{y}, \mathbf{y}) - 2 \cdot k(\mathbf{x}, \mathbf{y})$.

[2] The associated similarity matrix can be obtained by double centering [22] of the (squared) dissimilarity matrix $\mathbf{D}$: $\mathbf{S} = -\mathbf{J}\mathbf{D}\mathbf{J}/2$ with $\mathbf{J} = (\mathbf{I} - \mathbf{1}\mathbf{1}^\top/N)$ and identity matrix $\mathbf{I}$ and vector of ones $\mathbf{1}$.

ACCEPTED MANUSCRIPT

2.3. Indefinite Fisher and kernel quadratic discriminant

In [4, 5] the indefinite kernel Fisher discriminant analysis (iKFD) and indefinite kernel quadratic discriminant analysis (iKQD) was proposed focusing on binary classification problems, recently extended by a weighting scheme in [24][3].

The initial idea is to embed the training data into a Krein space (see Def. 2) and to apply a modified kernel Fisher discriminant analysis or kernel quadratic discriminant analysis for indefinite kernels. Consider binary classification and a data set of input-target training pairs $D = \{\mathbf{x}_i, y_i\}_{i=1}^N$, where $y_i \in \{-1, +1\}$. Given the indefinite kernel matrix K and the embedded data in a pseudo-Euclidean space (pE), the linear Fisher Discriminant function $f(\mathbf{x}) = \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle_{pE} + b$ is based on a weight vector $\mathbf{w}$ such that the between-class scatter is maximized while the within-class scatter is minimized along w. The dot product in pE is defined in Def. 2. $\Phi(\mathbf{x})$ is a vector of basis function evaluations for data item $\mathbf{x}$ and b is a bias term. This direction is obtained by maximizing the Fisher criterion in the pseudo-Euclidean space:

$$J(\mathbf{w}) = \frac{\langle \mathbf{w}, \Sigma_{pE}^b \mathbf{w} \rangle_{pE}}{\langle \mathbf{w}, \Sigma_{pE}^w \mathbf{w} \rangle_{pE}}$$

where $\Sigma_{pE}^b = \Sigma_b J$ is the scatter matrix in the pseudo-Euclidean space, with $J = diag(\mathbf{1}_p, -\mathbf{1}_q)$, where $\mathbf{1}_n \in \mathbb{R}^n$ denotes the n-dimensional vector of all ones. The *within-scatter-matrix* in the pseudo-Euclidean space is given as $\Sigma_{pE}^w = \Sigma_w J$. The *Euclidean* between- and within-scatter-matrices can be expressed as:

$$\Sigma_b = (\mu_+ - \mu_-)(\mu_+ - \mu_-)^\top \tag{2}$$

$$\Sigma_w = \frac{1}{|I_+|} \sum_{i \in I_+} (\phi(\mathbf{x}_i) - \mu_+)(\phi(\mathbf{x}_i) - \mu_+)^\top + \frac{1}{|I_-|} \sum_{i \in I_-} (\phi(\mathbf{x}_i) - \mu_-)(\phi(\mathbf{x}_i) - \mu_-)^\top, \tag{3}$$

where the set of indices of each class are $I_+ := \{i : y_i = +1\}$ and $I_- := \{i : y_i = -1\}$ and μ_+ and μ_- are the class-conditional means estimated on I_+ and I_-, respectively. To avoid the explicit embedding into the pE space (denoted as $\mathbb{R}^{(p,q)}$) a kernelization is

[3] For multiclass problems a classical 1-vs-rest wrapper is used within this paper

ACCEPTED MANUSCRIPT

considered such that the weight vector $\mathbf{w} \in \mathbb{R}^{(p,q)}$ is expressed as a linear combination of the training data: $\mathbf{w} = \sum_{i=1}^{N} \alpha_i \phi(\mathbf{x}_i)$. A similar strategy can be used for KQD as well as the indefinite kernel PCA [5].

2.4. Probabilistic Classification Vector Learning

Probabilistic Classification Vector Machine (PCVM) uses a kernel regression model $\sum_{i=1}^{N} w_i \phi_i(\mathbf{x}) + b$ with a link function, with w_i being again the weights of the basis functions $\phi_i(\mathbf{x})$ and b as a bias term. Unlike in the kernelized Fisher discriminant method described above, in PCVM the basis functions ϕ_i are defined explicitly as part of the model design. The Expectation Maximization (EM) implementation of PCVM [25] uses the probit link function, i.e. $\Psi(\mathbf{x}) = \int_{-\infty}^{x} \mathcal{N}(t|0,1)dt$, where $\Psi(\mathbf{x})$ is the cumulative distribution of the normal distribution $\mathcal{N}(0,1)$. We get: $l(\mathbf{x}; \mathbf{w}, b) = \Psi\left(\sum_{i=1}^{N} w_i \phi_i(\mathbf{x}) + b\right) = \Psi\left(\Phi(\mathbf{x})^\top \mathbf{w} + b\right)$

In the PCVM formulation [6], a truncated Gaussian prior with support on $[0, \infty)$ and mode at 0 is introduced for each weight w_i and a zero-mean Gaussian prior is adopted for the bias b. The priors are assumed to be mutually independent. $p(\mathbf{w}|\alpha) = \prod_{i=1}^{N} p(w_i|\alpha_i) \quad p(b|\beta) = \mathcal{N}(b|0, \beta^{-1})$, where

$$p(w_i|\alpha_i) = \begin{cases} 2\mathcal{N}(w_i|0, \alpha_i^{-1}) & \text{if } y_i w_i > 0 \\ 0 & \text{otherwise.} \end{cases}$$

We follow the standard probabilistic formulation and assume that $z(\mathbf{x}) = \Phi(\mathbf{x})^\top \mathbf{w} + b$ is corrupted by an additive random noise ϵ , where $\epsilon \sim \mathcal{N}(0,1)$. According to the probit link model, we have:

$$\begin{aligned} h(\mathbf{x}) &= \Phi(\mathbf{x})^\top \mathbf{w} + b + \epsilon \geq 0, \qquad \text{if } y = 1, \\ h(\mathbf{x}) &= \Phi(\mathbf{x})^\top \mathbf{w} + b + \epsilon < 0, \qquad \text{if } y = -1 \end{aligned} \tag{4}$$

and obtain:

$$p(y = 1|\mathbf{x}, \mathbf{w}, b) = p(\Phi(\mathbf{x})^\top \mathbf{w} + b + \epsilon \geq 0) = \Psi(\Phi(\mathbf{x})^\top \mathbf{w} + b).$$

Note that $h(\mathbf{x})$ is a latent variable because ϵ is an unobservable variable. We collect evaluations of $h(\mathbf{x})$ at training points in a vector $\mathbf{H}(\mathbf{x}) = (h(\mathbf{x_1}), \ldots, h(\mathbf{x_N}))^\top$. In the

ACCEPTED MANUSCRIPT

expectation step the expected value $\bar{\mathbf{H}}$ of $\mathbf{H}$ with respect to the posterior distribution over the latent variables is calculated (given old values $\mathbf{w}^{\text{old}}, b^{\text{old}}$). In the maximization step the parameters are updated through

$$\mathbf{w}^{\text{new}} = M(M\Phi^{\top}(\mathbf{x})\Phi(\mathbf{x})M + I_N)^{-1}M(\Phi^{\top}(\mathbf{x})\bar{\mathbf{H}} - b\Phi^{\top}(\mathbf{x})\mathbf{1}) \quad (5)$$

$$\mathbf{b}^{\text{new}} = t(1+tNt)^{-1}t(\mathbf{1}^{\top}\bar{\mathbf{H}} - \mathbf{1}^{\top}\Phi(\mathbf{x})^{\top}\mathbf{w}) \quad (6)$$

where I_N is a N-dimensional identity matrix and $\mathbf{1}$ a all-ones vector, the diagonal elements in the diagonal matrix M are:

$$M_{ii} = (\bar{\alpha}_i)^{-1/2} = \begin{cases} \sqrt{2}w_i & \text{if} \quad y_i w_i \geq 0 \\ 0 & \text{else} \end{cases} \quad (7)$$

and the scalar $t = \sqrt{2}|b|$. Further details can be found in [6]. Even though kernel machines and their derivatives have shown great promise in practical application, their scope is somehow limited by the fact that the computational complexity grows rapidly with the size of the kernel matrix (number of data items). Among methods suggested to deal with this issue in the literature, the Nyström method has been popular and widely used.

3. Nyström approximated matrix processing

The Nyström approximation technique has been proposed in the context of kernel methods in [13]. Here, we give a short review of this technique before it is employed in PCVM and iKFD. One well known way to approximate a $N \times N$ Gram matrix, is to use a low-rank approximation. This can be done by computing the eigendecomposition of the kernel matrix $K = U\Lambda U^T$, where U is a matrix, whose columns are orthonormal eigenvectors, and Λ is a diagonal matrix consisting of eigenvalues $\Lambda_{11} \geq \Lambda_{22} \geq ... \geq 0$, and keeping only the m eigenspaces which correspond to the m largest eigenvalues of the matrix. The approximation is $\tilde{K} \approx U_{(N,m)}\Lambda_{(m,m)}U_{(m,N)}$, where the indices refer to the size of the corresponding submatrix restricted to the larges m eigenvalues. The Nyström method approximates a kernel in a similar way, without computing the eigendecomposition of the whole matrix, which is an $O(N^3)$ operation.

ACCEPTED MANUSCRIPT

By the Mercer theorem, kernels $k(\mathbf{x}, \mathbf{x}')$ can be expanded by orthonormal eigenfunctions φ_i and non negative eigenvalues λ_i in the form

$$k(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^{\infty} \lambda_i \varphi_i(\mathbf{x}) \varphi_i(\mathbf{x}').$$

The eigenfunctions and eigenvalues of a kernel are defined as solutions of the integral equation

$$\int k(\mathbf{x}', \mathbf{x}) \varphi_i(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} = \lambda_i \varphi_i(\mathbf{x}'),$$

where $p(\mathbf{x})$ is a probability density over the input space. This integral can be approximated based on the Nyström technique by an i.i.d. sample $\{\mathbf{x}_k\}_{k=1}^m$ from $p(\mathbf{x})$:

$$\frac{1}{m} \sum_{k=1}^{m} k(\mathbf{x}', \mathbf{x}_k) \varphi_i(\mathbf{x}_k) \approx \lambda_i \varphi_i(\mathbf{x}'). \tag{8}$$

Using this approximation we denote with $K^{(m)}$ the corresponding $m \times m$ Gram submatrix and get the corresponding matrix eigenproblem equation as:

$$\frac{1}{m} K^{(m)} U^{(m)} = U^{(m)} \Lambda^{(m)}$$

with $U^{(m)} \in \mathbb{R}^{m \times m}$ is column orthonormal and $\Lambda^{(m)}$ is a diagonal matrix.

Now we can derive the approximations for the eigenfunctions and eigenvalues of the kernel k

$$\lambda_i \approx \frac{\lambda_i^{(m)} \cdot N}{m}, \quad \varphi_i(\mathbf{x}') \approx \frac{\sqrt{m/N}}{\lambda_i^{(m)}} \mathbf{k}_x'^{\top} \mathbf{u}_i^{(m)}, \tag{9}$$

where $\mathbf{u}_i^{(m)}$ is the ith column of $U^{(m)}$. Thus, we can approximate φ_i at an arbitrary point $\mathbf{x}'$ as long as we know the vector $\mathbf{k}_x' = (k(\mathbf{x}_1, \mathbf{x}'), ..., k(\mathbf{x}_m, \mathbf{x}'))$. For a given $N \times N$ Gram matrix K one may randomly choose m rows and respective columns. The corresponding indices are called landmarks, and should be chosen such that the data distribution is sufficiently covered. Strategies how to chose the landmarks have recently been addressed in [14, 26] and [27, 16]. We denote these rows by $K_{(m,N)}$. Using the formulas Eq. (9) we can reconstruct the original kernal matrix,

$$\tilde{K} = \sum_{i=1}^{m} 1/\lambda_i^{(m)} \cdot K_{(m,N)}^T (\mathbf{u}_i^{(m)})^T (\mathbf{u}_i^{(m)}) K_{(m,N)},$$

ACCEPTED MANUSCRIPT

where $\lambda_i^{(m)}$ and $\mathbf{u}_i^{(m)}$ correspond to the $m \times m$ eigenproblem (8). Thus we get the approximation,

$$\tilde{K} = K_{(N,m)} K_{(m,m)}^{-} K_{(m,N)}. \tag{10}$$

This approximation is exact, if $K_{(m,m)}$ has the same rank as K.

3.1. Pseudo Inverse and Singular Value Decomposition of a Nyström approximated matrix

In the Ny-PCVM approach discussed in Section 5 we need a inverse of a Nyström approximated matrix, while for the Ny-iKFD a Nyström approximated eigenvalue decomposition (EVD) is needed.

A Nyström approximated inverse can be regularized by a modified singular value decomposition (SVD) with a rank limited by $r^* = \min\{r, m\}$, where r is the rank of the obtained inverse and m the number of landmark points. The output is given by the rank reduced left and right singular vectors and the reciprocal of the singular values. The singular value decomposition based on a Nyström approximated similarity matrix $\tilde{K} = K_{(N,m)} K_{(m,m)}^{-1} K_{(N,m)}^{\top}$ with m landmarks, calculates the left singular vectors of $\tilde{K}$ as the eigenvectors of $\tilde{K}\tilde{K}^{\top}$ and the right singular vectors of $\tilde{K}$ as the eigenvectors of $\tilde{K}^{\top}\tilde{K}$[4]. The r^* non-zero singular values of $\tilde{K}$ are then found as the square roots of the non-zero eigenvalues of both $\tilde{K}^{\top}\tilde{K}$ or $\tilde{K}\tilde{K}^{\top}$. Accordingly, one only has to calculate a new Nyström approximation of the matrix $\tilde{K}\tilde{K}^{\top}$ using e.g. the same landmark points as for the input matrix $\tilde{K}$. Subsequently an eigenvalue decomposition (EVD) is calculated on the approximated matrix $\zeta = \tilde{K}\tilde{K}^{\top}$. For a matrix approximated by Eq. (10) it is possible to compute its exact eigenvalue estimators in linear time[5].

3.2. Eigenvalue decomposition of a Nyström approximated matrix

To compute the eigenvectors and eigenvalues of an *indefinite* matrix we first compute the squared form of the Nyström approximated kernel matrix. Let K be a psd

[4] For symmetric matrices we have $\tilde{K}\tilde{K}^{\top} = \tilde{K}^{\top}\tilde{K}$

[5] assuming $m \ll N$, in particular $m < N^{1/3}$

ACCEPTED MANUSCRIPT

similarity matrix, for which we can write its decomposition as

$$\tilde{K} = K_{(N,m)} K_{(m,m)}^{-1} K_{(m,N)} = K_{(N,m)} U \Lambda^{-1} U^\top K_{(N,m)}^\top = BB^\top,$$

where we defined $B = K_{(N,m)} U \Lambda^{-1/2}$ with U and Λ being the eigenvectors and eigenvalues of $K_{(m,m)}$, respectively.

Further it follows for the *squared* $\tilde{K}$:

$$\tilde{K}^2 = BB^\top BB^\top = BVAV^\top B^\top,$$

where V and A are the eigenvectors and eigenvalues of $B^\top B$, respectively. The square operation does not change the eigenvectors of K but only the eigenvalues. The corresponding eigenequation can be written as $B^\top Bv = av$. Multiplying with B from left we get:

$$\underbrace{BB^\top}_{\tilde{K}} \underbrace{(Bv)}_{u} = a \underbrace{(Bv)}_{u}.$$

It is clear that A must be the matrix with the eigenvalues of $\tilde{K}$. The matrix Bv is the matrix of the corresponding eigenvectors, which are orthogonal but not necessary orthonormal. The normalization can be computed from the decomposition:

$$\tilde{K} = B \underbrace{VV^\top}_{I} B^\top = BVA^{-1/2} AA^{-1/2} V^\top B^\top = CAC^\top,$$

where we defined $C = BVA^{-1/2}$ as the matrix of orthonormal eigenvectors of K. The eigenvalues of $\tilde{K}$ can be obtained using $A = C^\top \tilde{K} C$. Using this derivation we can obtain exact eigenvalues and eigenvectors of an indefinite low rank kernel matrix K, given $\text{rank}(K) = m$ and the landmarks points are independent[6]

The accuracy of this approximation is typically measured by the Frobenius norm. A low value of the Frobenius norm of the approximated versus the original kernel matrix ensures that the approximated kernel matrix $\tilde{K}$ can be used instead of K for any kernel based data analysis method, such as kernel-PCA, kernel-k-means, SVM, Laplacian

[6] An implementation of this linear time eigen-decomposition for low rank indefinite matrices is available at: `http://www.techfak.uni-bielefeld.de/~fschleif/eigenvalue_corrections_demos.tgz`.

ACCEPTED MANUSCRIPT

eigenmaps. In the context of classification the requirement of close approximation of the kernel matrix may be too strong and unnecessary. After all, a low rank kernel matrix which preserves class separation is sufficient. To achieve this objective we suggest to use a *supervised* landmark selection scheme introduced in the following section.

4. Supervised landmark selection using minimum enclosing balls

The original (unsupervised) Nyström approximation is based on m characteristic landmark points taken from the dataset. The number of landmarks should be sufficiently large and the landmarks should be diverse enough to get accurate approximations of the dominating singular vectors of the similarity matrix. In [14] multiple strategies for landmark selection have been studied and a clustering based approach was suggested to find the specific landmarks. Thereby the number of landmarks is a user defined parameter and a classical k-means algorithm is applied on the kernel matrix to identify characteristic landmark points in the empirical feature space. This approach is quite effective (see [14]), with some small improvements using an advanced clustering scheme as shown in [15]. Other recent proposals along those lines, e.g. leverage scores [17], are much more costly with at least quadratic costs and therefore not applicable in our setting. We will use the k-means approach as a baseline for an advanced landmark section approach. Further, we will also consider a pure random selection strategy as another baseline. It should be noted that the formulation given in [14] takes the full kernel matrix as an input into the k-means clustering. This is obviously also very costly and may become inapplicable for larger kernel matrices [7]

In general, the approaches discussed above only address the problem of the selection or *positioning* of the landmarks, given their number. It is not clear how the *number* of landmarks can be appropriately chosen. Clearly, if the number of landmarks is large, we can expect the data space to be sufficiently covered, but the model complexity can become prohibitive. On the other hand, if the number of landmarks is too small, the kernel matrix approximation may be poor.

[7] It may however be possible to circumvent this full complexity approach e.g. by subsampling concepts or by more advanced concepts of k-means, but this is not the focus of this paper.

ACCEPTED MANUSCRIPT

We propose to consider the Nyström approximation in a restricted form with respect to a *supervised* learning problem. This relieves us from the need of a perfect reconstruction of the kernel matrix. It is in fact sufficient to reconstruct the kernel such that it is close to the ideal kernel (see e.g. [28]). We will however not learn an idealized kernel as proposed in [28], which by itself is very costly for large scale matrices, but provide a landmark selection strategy motivated by similar intuitions.

The (supervised) representation accuracy of the Nyström approximation of K depends on the number of the selected landmarks and the used landmark selection scheme. We propose to calculate minimum enclosing ball solutions (MEB) on the individual class-wise kernel matrices. This will enable us to

1. find a sufficient number of landmarks for the given classification task,
2. find landmark positions preserving a good class separation.

Note that the chosen landmarks may not necessarily lead to a good reconstruction of $\hat{K}$, as measured e.g. by the Frobenius norm. As an additional constraint we are looking for an approach where also indefinite proximity matrices can be processed without costly preprocessing steps.

4.1. MEB for psd input kernels

We denote the set of indices or points of a sub kernel matrix referring to class j by Ω_j. Assuming approximately spherical classes (in the feature space), we invoke the **minimum enclosing ball** method on each class separately:

$$\begin{aligned} &\min_{R^2, \mathbf{w}_j} \quad R^2 \\ &\text{such that} \quad \|\mathbf{w}_j - \Phi(\xi_i)\|^2 \leq R^2 \quad \forall \xi_i \in \Omega_j \end{aligned}$$

where R is the radius of the sphere and $\mathbf{w}_j$ is a center of class j, which can be indirectly represented in the kernel space as a weighted linear combination of the points in Ω_j. The assumption of a sphere is in fact no substantial restriction if the provided kernel is sufficiently "expressive". This is also the reason why core-vector data description (CVDD) can be used as a linear time replacement for support vector data description [29].

ACCEPTED MANUSCRIPT

It has been shown e.g. in [30] that the minimum enclosing ball can be approximated with quality $\epsilon > 0$ in (worst case) linear time using an algorithm which requires only a constant subset of Ω_j, the core set. Given ϵ, the following algorithm converges in $\mathcal{O}(1/\epsilon^2)$ steps:

MEB:

Choose $\xi_i \in \Omega_j$ randomly. Find $\xi_k \in \Omega_j$ furthest away from ξ_i in the feature space (e.g. maximizing $\|\Phi(\xi_i) - \Phi(\xi_k)\|^2$). $S := \{\xi_i, \xi_k\}$.

repeat

solve MEB$(S) \rightarrow \tilde{\mathbf{w}}_j, R$

if there is $\xi_l \in \Omega_j$ with $\|\Phi(\xi_l) - \tilde{\mathbf{w}}_j\|^2 > R^2(1+\epsilon)^2$ **then**

$S := S \cup \{\xi_l\}$

end if

until all ξ_l are covered by the $R(1+\epsilon)$ ball in the feature space

return $\tilde{\mathbf{w}}_j$

In each step, the MEB problem is solved for a small subset of constant size only. This is possible by referring to the dual problem which has the form

$$\begin{array}{ll} \min_{\alpha_i \geq 0} & \sum_{ij} \alpha_i \alpha_j K_{ij} - \sum_i \alpha_i K_{ii}^2 \\ \text{where} & \sum_i \alpha_i = 1 \end{array}$$

with operations only involving dot products, i.e. kernelization is possible. The same holds for all distance computations of the approximate MEB problem. Note that the dual MEB problem provides a solution in terms of the dual variables α_i. The identified finite number of core points (those with non-vanishing α_i) will be used as landmarks for this class and considered to be sufficient to represent the enclosing sphere of the data. Each class is represented by at least two core points. Combining all core sets of the various classes provides us with the full set of landmarks used to get a Nyström approximation of K.

The MEB solution typically consists of a very small number of points (independent of N), sufficient to describe the hyper-ball enclosing the respective data. If the kernel is psd we can use the MEB approach directly in the kernel space.

ACCEPTED MANUSCRIPT

Algorithm 1 Proposed handling of indefinite kernels by the MEB approach

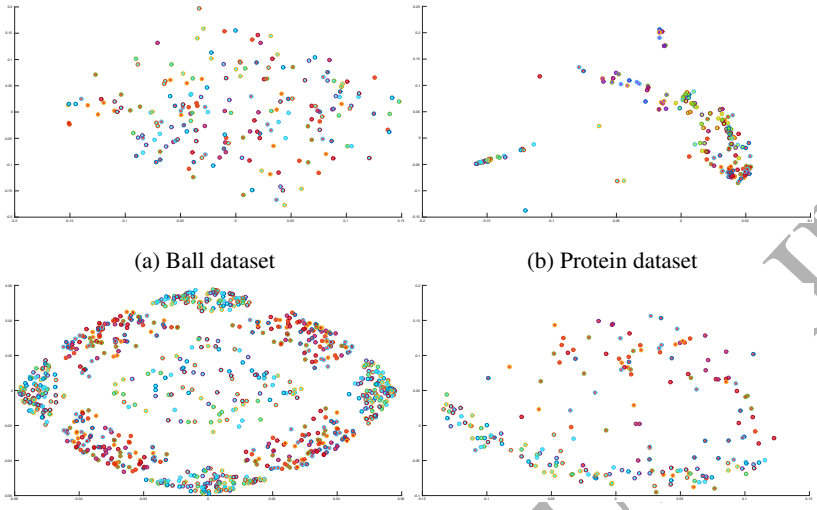
1. let $k(\mathbf{x}, \mathbf{x}')$ be a symmetric (indefinite) similarity function (e.g. a sequence alignment)
2. for all classes j let $\Omega_j = \{\mathbf{x}_i : y_i = j\}$
3. calculate the (indefinite) kernel matrix K_j using Ω_j and $k(\mathbf{x}, \mathbf{x}')$
4. if the kernel matrix is indefinite, apply a square operation on the small matrix K_j by using $K_j \cdot K_j^\top$
5. apply the MEB algorithm for each of the kernel matrices K_j with $\epsilon = 0.01$
6. combine all landmark indices obtained from the previous step and calculate the Nyström approximation using Eq. (10)
7. apply Ny-PCVM or Ny-iKFD using the approximated kernel matrix

4.2. MEB for non-psd input kernels

If the given kernel is non-psd we either can apply various eigenvalue correction approaches see [10], or we use $\hat{K} = K \cdot K^\top$, which can also be easily done for Nyström approximated matrices without calculating a full matrix (see first part of Eq. (15)). This procedure does not change the eigenvectors of K but takes the square of the eigenvalues such that $\hat{K}$ becomes psd. It should be noted that if we use $\hat{K}$ as an input of a kernel k-means algorithm this is equivalent as using K as the input of the classical k-means with Euclidean distance as suggested in [14].

The proposed supervised landmark selection using MEB does not only identify an estimate for the number of landmarks, but it also suggests their position. The solutions of the MEB consist of non-redundant points at the perimeter of the sphere, which can considered to be unrelated, although not necessarily orthogonal in the feature space (with potentially squared negative eigenvalues). Especially only those points are included in the MEB solution which are needed to explain the sphere such that redundancy within this set is avoided [30]. We will show the effectiveness of this approach in some short experiments. A pseudo code of the suggested algorithm is given in Alg. 1.

ACCEPTED MANUSCRIPT



(a) Ball dataset (b) Protein dataset

(c) Checker dataset (d) Gauss dataset

Figure 1: Laplacian eigenmap visualization of the initial test and simulated similarity matrices using $K \cdot K^\top$. Colors/shades indicate the different classes. Axis labeling is arbitrary.

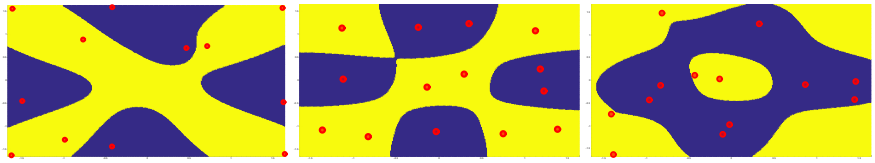
4.3. Small scale experiments - landmark selection scheme

We use the ball dataset as proposed in [31]. It is an artificial dataset based on the surface distances of randomly positioned balls of two classes having a slightly different radius. The dataset is non-Euclidean with substantial information encoded in the negative part of the eigenspectrum. We generated the data with 100 samples per class leading to an $N \times N$ dissimilarity matrix $\mathbf{D}$, with $N = 200$.

We also use the protein data (213 pts, 4 classes) set represented by an indefinite similarity matrix, with a high intrinsic dimension [10]. Further we analyzed two simulated metric datasets which are not linear separable using the Euclidean norm: (1) the checker board data, generated as a two dimensional dataset with datapoints organized on a 3×3 checkerboard, with alternating labels. This dataset has *multi-modal* classes. (2) a simple Gaussian cloud dataset with two Gaussian with substantial overlap. The simulated data have been represented by an extreme learning machine (elm) kernel. Checker is linear separable in the elm-kernel space, whereas Gaussian is not separable by construction.

ACCEPTED MANUSCRIPT IPT

ACCEPTED MANUSCRIPT



(a) Checker board data with the MEB selection scheme.

(b) Checker board data with the k-means selection scheme using #MEB landmarks.

(c) Checker board data with the random selection scheme using #MEB landmarks.

Figure 2: Typical plots of the checker board data - taken from the crossvalidation models - with iKFD predictions using different landmark selection schemes and an elm kernel. The worst result $\approx 72\%$ is obtained by plot c) using the random sampling strategy whereby the number of landmarks was chosen from the MEB approach. The selected landmark points are indicated as (red) circles. In plot b) one clearly sees that k-means has rearranged the points to cover the whole data space. For the random approach we observe that some points are very close to each other (and have the same label) and are therefore not very informative. The MEB solution in plot a) leads to very good prediction results on the test data with around 90%, which is only slightly worse than the result for b) with 92%.

It should be noted that the elm kernel, used for the vectorial data, typically increases the number of non-vanishing eigenvalues such that the original two dimensional data are finally indeed higher dimensional and not representable by only two basis functions. Two dimensional visualizations of the unapproximated $K \cdot K^\top$ similarity matrices obtained by using Laplacian eigenmaps [32]. are shown in Figure 1. For the checker board data we also show two-dimensional plots of the obtained iKFD decision boundaries and different landmark selection schemes in Figure 2.

Now the obtained (indefinite) kernel matrix has been used in the iKFD in six different ways using different landmark selection schemes:

a) we used the original kernel matrix (SIM1),

b) the matrix is Nystöm approximated using the MEB approach (SIM2),

c) the matrix is Nyström approximated using the approach of [14] where the number of landmarks is taken from the MEB solution (SIM3),

d) using the approach of [14] but with C landmarks where C is the number of classes (SIM4)

ACCEPTED MANUSCRIPT

e) using a random sample of C landmarks (SIM5). SIM5 can be considered as a very basic baseline approach.

f) using an entropy based selection as proposed in [16] (SIM6) [8] where the number of landmarks is again taken from the MEB solution

One may also simply use a *very* large number of randomly selected landmarks, but this can become prohibitive if N is large such that the calculation of $N \times m$ similarities can be costly in memory and runtime. Further it can be very unattractive to have a larger m for the out of sample extension to new points. If for example costly alignment scores are used one is interested in having a very small m to avoid large costs in the test phase of the model.

The results of a 10-fold crossvalidation are shown in the Table 1 with runtimes given in Table 2. Here and in the following experiments the landmark selection was part of the crossvalidation scheme and the landmarks are selected on the training set only and the test data have been mapped to the approximated kernel space by the Nyström kernel expansion (see e.g. [13]).

For the ball data set the data contain substantial information in the negative fraction of the eigenspectrum, accordingly one may expect that these eigenvalues should not be removed. This is also reflected in the results. In SIM4 and SIM 5 only the two dominating eigenvectors are kept such that the negative eigenvalues are removed, degenerating the prediction accuracy. The SIM3 encoding is a bit better, but the landmark optimization via k-means is not very effective for this dataset. Also the entropy approach in SIM6 was not very efficient. The SIM2 encoding has a substantial drop in the accuracy with respect to the unapproximated kernel but the intrinsic dimension of the dataset is very high and the $m = 8$ landmarks are enough to preserve the dominating positive and negative eigenvalues. The unapproximated kernel leads to perfect separation, clearly showing that the negative eigenspectrum contains discriminative information. The respective eigenvalue plots are provided in Figure 3.

[8] We use the implementation as provided by the authors in the LSSVM toolbox `http://www.esat.kuleuven.be/sista/lssvmlab/`

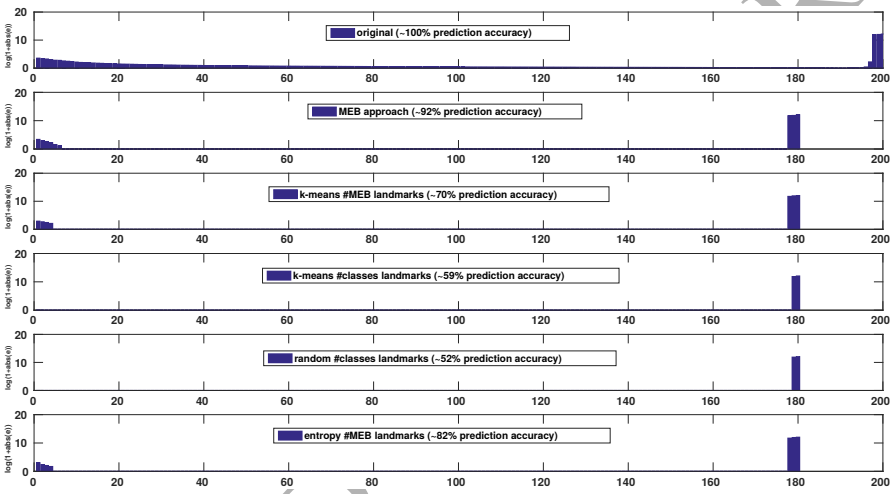


Figure 3: Eigenvalue analysis of the ball dataset using the different approaches. The first plot shows the eigenvalues of the original kernel (SIM1), the other plots show typical results from the 10-fold crossvalidation for the various landmark selection approaches (SIM2-SIM6). It can be clearly seen that the landmarks identified by the MEB approach sufficiently capture the negative eigenvalues. The random sampling approach works only if a larger number of landmarks is chosen and is still less efficient because it is not ensured that the landmarks cover the whole data space. Especially if the data are non i.i.d. random sampling is typically insufficient.

ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

Table 1: Test set results of a 10-fold iKFD run on the simulated / controlled datasets in different kernel approximations. A $\star$ indicates a non-metric similarity matrix. The number of identified landmarks is shown in brackets for SIM2.

Method	Ball$\star$	Protein$\star$	Checker	Gaussian
$SIM1\|s(\hat{K}, K)$	100 ± 0	98.12 ± 3.22	98.89 ± 0.35	90.00 ± 5.77
$SIM2\|s(\hat{K}, K)$	$92.00 \pm 4.83(8)$	$96.71 \pm 3.20(25)$	$90.22 \pm 8.52(9)$	$90.00 \pm 7.45(8)$
$SIM3\|s(\hat{K}, K)$	70.00 ± 12.69	96.71 ± 4.45	91.78 ± 9.24	87.00 ± 10.33
$SIM4\|s(\hat{K}, K)$	59.50 ± 5.50	86.85 ± 6.29	65.33 ± 5.13	65.00 ± 8.17
$SIM5\|s(\hat{K}, K)$	52.50 ± 12.08	78.87 ± 14.61	46.11 ± 4.20	77.50 ± 10.61
$SIM6\|s(\hat{K}, K)$	74.50 ± 12.79	95.31 ± 5.78	62.33 ± 11.67	87.00 ± 7.52

The results show that the proposed MEB approach is capable in preserving the geometric information also for the negative (squared) eigendimensions while being quite simple. We believe that controlling the approximation accuracy of the kernel by ϵ in the MEB is much easier than selecting the number of clusters (per class) in k-means clustering. In fact it will almost always be sufficient to keep $\epsilon \approx 0.01$ to get reliable landmark sets whereas the number of clusters is very dataset dependent and not easy to choose. However, in contrast to the results shown in Table 1 the approach by [14] is typically effective for a large variety of datasets also with indefinite kernels, given the number of landmarks is reasonable large and discriminating information is sufficiently provided in the dominating eigenvectors of the cluster solutions. For the protein data we observe similar results and the proposed approach, the k-means strategy and the entropy approach are effective. SIM4 and SIM5 is again substantially worse because four landmarks are in general not sufficient to represent these data from a discriminative point of view.

For the checker board and Gaussian data SIM2 and SIM3 are again close and SIM4 and SIM5 are substantially worse using only two landmark points. The entropy approach was efficient only for the Gaussian data, but failed for Checker which may be attributed to the strong multi-modality of the data.

The runtimes given, in Table 2, show already for the small data examples that the MEB approach is much faster then k-means or the entropy approach if the number of points gets larger which was already expected from the theoretical runtime complexity of these algorithms.

ACCEPTED MANUSCRIPT

Table 2: Runtimes in seconds of a 10-fold iKFD run on the simulated / controlled datasets in different kernel approximations. A $\star$ indicates a non-metric similarity matrix.

Method	Ball$\star$	Protein$\star$	Checker	Gaussian
$SIM1\|s(\hat{K}, K)$	0.5	0.82	13.45	0.74
$SIM2\|s(\hat{K}, K)$	1.0	1.56	3.76	0.98
$SIM3\|s(\hat{K}, K)$	1.57	2.57	14.77	1.51
$SIM4\|s(\hat{K}, K)$	0.84	1.14	13.23	0.90
$SIM5\|s(\hat{K}, K)$	0.61	0.98	3.23	0.65
$SIM6\|s(\hat{K}, K)$	3.2	8.47	8.12	3.94

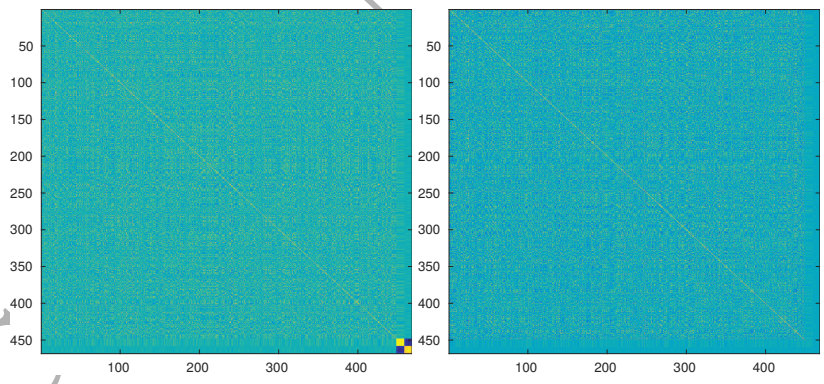


Figure 4: Reconstructed kernel matrix (from the crossvalidation run) of the 10 dimensional Gaussian example. Left using the MEB approach, right using the k-means landmark selection. Note the small region on the bottom in the left plot indicating the smaller Gaussian which are almost missing in the right plot.

ACCEPTED MANUSCRIPT

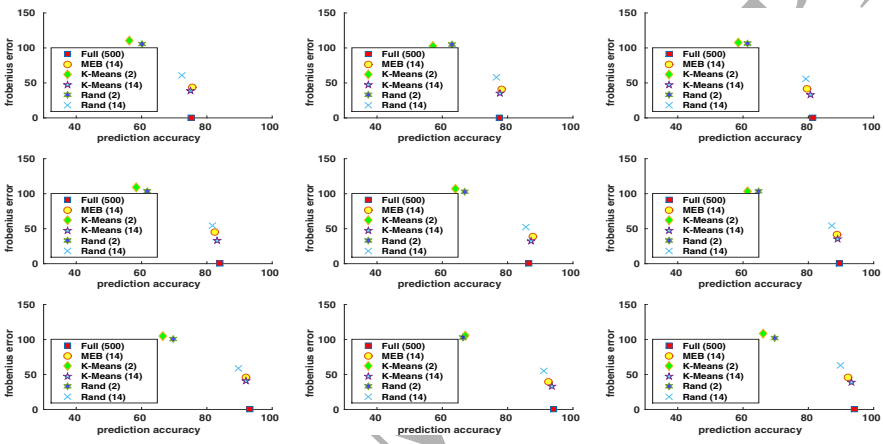


Figure 5: Results for the different landmark selection schemes on a dataset of two banana like shaped distributions with varying overlap (from strong overlap - top left, to almost no overlap and good separation right, bottom). We see that the prediction accuracy is improving with better separation of the distributions. One can also see that a random selection of one landmark per class fails. If the number of landmarks is chosen more appropriately (by using the number as obtained from the MEB solution) the accuracy improves but is still worse for a random selection approach. If the landmarks are optimized using k-means the Frobenius error typically shrinks but the accuracy is not substantially effected. The MEB approach shows consistently good prediction error, although a slightly higher Frobenius error. We clearly see that a higher Frobenius error may *not* lead to a high prediction error.

ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

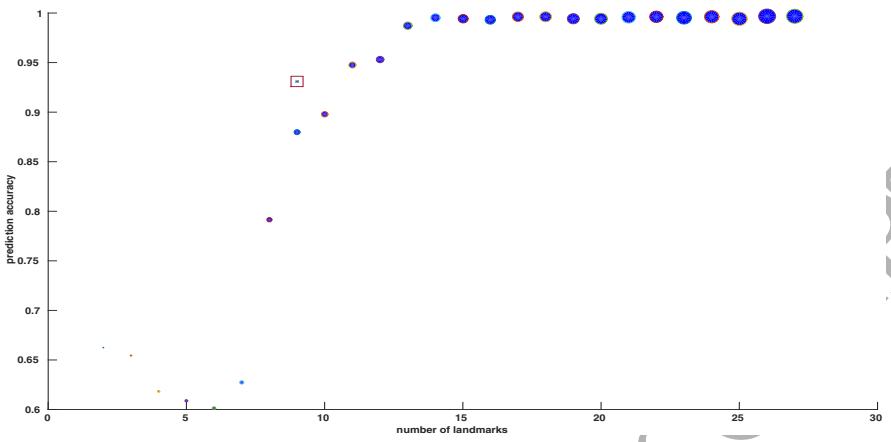


Figure 6: Prediction accuracy for the checker board data with a varying number of landmarks. The MEB solution is indicated by a square. The x axis has an increasing number of landmarks and the y-axis shows the respective prediction error from the crossvalidation using a k-means based Nyström approximation. We see that the MEB solution is almost optimal and a further increase of the number of landmarks has only a small effect. The Frobenius error is inversely scaled such that a low Frobenius error is shown by a large ball

In Figure 5 we analyze a dataset with two banana shaped distributions and varying overlap for the different landmark selection schemes. Initially we only know that we have two classes, so we may conclude that we have two clusters and hence it maybe sufficient to consider two landmarks, only. As the plot shows this is not a very good strategy and works only somewhat if the data are very well separated (right, bottom subplot). If the data show overlap it is helpful to have a more advanced selection strategy. We see that MEB provides a good choice for the number of landmarks and in general leads to very good prediction results, although the Frobenius error maybe higher. K-means will in general improve the Frobenius error but has still some errors if the number of landmarks (or in k-means clusters) is not well determined ($\star$). Only with a good pre-condition using the number of landmarks suggested by MEB ($\diamond$), the k-means gives very good results, with low Frobenius error.

In Figure 6 we consider again the checker board data but by varying the number of landmarks. The Nyström approximation was done by k-mean where the number of landmarks was given and stepwise increased for each result. We can see, that the MEB solution not only has given a good estimate for a reasonable number of landmarks, but

ACCEPTED MANUSCRIPT

has also directly provided a reliable good matrix approximation. Additional landmarks have only a minor effect on the prediction accuracy, but help to improve the Frobenius error.

In another small experiment we analyzed the effect of the k-means based landmark selection [14] in more detail. We consider three Gaussians where one Gaussian has 500 points spread in two dimensions and two other Gaussians each with 20 points spread in another dimensions. All Gaussians are perfectly separated to each other located in a three dimensional space. To make the task more challenging we further add 7 dimensions with small noise contributions to the large Gaussian. The final data are given in a 10 dimensional space, whereby the small Gaussians are intrinsically low dimensional and the large Gaussian is 10 dimensional. with major contributions only in two dimensions. The points from the large Gaussian are labeled 0 and the other 1. Using the MEB approach we obtain 10 landmarks and the approximated kernel is sufficient to give a perfect prediction of 100% in a 10-fold crossvalidation with iKFD. Using the k-means or entropy based approach (with the same number of landmarks) the prediction accuracy drops down to $\approx 84\%$ and for random sampling we get a prediction accuracy in the same range of 83% - again with 10 landmarks . This can be explained by the behavior of k-means to assign the prototypes or landmarks to dense regions. It is hence more likely that after the k-means clustering (almost) all prototypes are used to represent the large Gaussian and no prototypes are left for the other classes. Due to the fact that the other classes are located in different dimensions with respect to the large Gaussian these dimensions are not any longer well represented and hence the respective classes are often missing in the approximated kernel (see Figure 4). This density related behavior is also known as magnification [33] in the context of different vector quantization approaches. Hence using the unsupervised k-means landmark selection it can easily happen, that the majority of the data space is well presented but small classes are ignored - which is obviously a problem for a supervised data analysis.

From these initial experiments we see that the proposed landmark selection scheme is sufficient to approximate the original kernel function for a *supervised* analysis as indicated by the prediction accuracy of the iKFD model. We also see that the Nyström approximation can introduce substantial error if the data are not low rank (for checker)

ACCEPTED MANUSCRIPT

due to a more complicated kernel mapping aka similarity function. We would like to highlight again that without an advocated guess of the number of landmarks neither the k-means strategy nor the entropy approach are very efficient.

In the experiment in section 7 we will restrict our analysis to the proposed landmark selection using the MEB approach, the k-means strategy and the entropy based technique.

5. Large scale indefinite learning with PCVM and iKFD

We now integrate the aforementioned Nyström approximation approaches and the supervised landmark selection into PCVM and iKFD. The modifications ensure that all matrices are processed with linear memory complexity and that the underlying algorithms have a linear runtime complexity. For both algorithms the initial input is the Nyström approximated kernel matrix with landmarks selected by using one of the formerly provided landmark selection schemes.

5.1. PCVM for large scale proximity data

The PCVM parameters are optimized using the EM algorithm to prune the weight vector $\mathbf{w}$ during learning and hence the considered basis functions representing the model. We will now show multiple modifications of PCVM to integrate the Nyström approximation and to ensure that the memory and runtime complexity remains linear at all time. We refer to our method as Ny-PCVM. Initially the Ny-PCVM algorithm makes use of the matrices $K_1 = K_{(N,m)}$ and $K_2 = K_{(m,m)}^{-1} \cdot K_1^\top$ obtained from the original kernel matrix using the Nyström landmark technique described above. Given a matrix X, we denote by $\hat{X}$ the matrix formed from X containing elements at indices that have not yet been pruned out of the weight vector $\mathbf{w}$. As an example, the matrices $\hat{K_1} = K_1^{\mathbf{w}\neq 0,\cdot}$, $\hat{K_2} = K_2^{\cdot,\mathbf{w}\neq 0}$ hold only those columns/rows of K_1 or K_2 not yet pruned out from the weight vector. We will use the same notation also for other variables. We denote the set of indices of m randomly selected landmarks by $[m]$. Finally, in contrast to the original PCVM formulation [6], in our notation we explicitly use the data labels - for example, instead of vector $\Phi_\theta(\mathbf{x})$ we write $\Xi_\theta(\mathbf{x}) \circ \mathbf{y}$, where $\Xi_\theta(\mathbf{x})$ is

ACCEPTED MANUSCRIPT

the kernel vector of $\mathbf{x}$ without any label information, $\mathbf{y}$ is the label vector and $\circ$ is the element-wise multiplication.

We now adapt multiple equations of the original PCVM to include the Nyström approximated matrix. Eq. (4) for the i-th training point now reads:

$$z_{i,\theta} = \Xi_\theta(\mathbf{x}_i)(\mathbf{y} \circ \mathbf{w}) + b, \tag{11}$$

in matrix notation for all training points:

$$\hat{\mathbf{z}} = (((\hat{\mathbf{y}} \circ \hat{\mathbf{w}})^\top \hat{K_1}) \cdot K_2)^\top + b. \tag{12}$$

We obtain column vectors $\bar{\mathbf{H}}_\theta$ and the reduced form $\bar{\hat{\mathbf{H}}}_\theta$, by using only the non-vanishing basis functions and the Nyström approximated matrices in Eq. (4). In the maximization step of the original PCVM the $\mathbf{w}$ are updated as (see Eq. (5)):

$$\mathbf{w}^{\text{new}} = M\underbrace{(M\Phi_\theta(\mathbf{x})^\top \Phi_\theta(\mathbf{x})M + I_N)}_{\Upsilon}{}^{-1} M(\Phi_\theta(\mathbf{x})^\top \bar{\mathbf{H}}_\theta - b\Phi_\theta(\mathbf{x})^\top \mathbf{1}) \tag{13}$$

To account for the now excluded labels we reformulate Equation (5) as:

$$\mathbf{w}^{\text{new}} = M\underbrace{(M(\Xi_\theta(\mathbf{x})^\top \Xi_\theta(\mathbf{x})\hat{\mathbf{y}}^\top \hat{\mathbf{y}})M + I_N)}_{\Upsilon}{}^{-1} M(\hat{\mathbf{y}}^\top(\Xi_\theta(\mathbf{x})^\top \bar{\mathbf{H}}_\theta) - b\hat{\mathbf{y}}^\top(\Xi_\theta(\mathbf{x})^\top \mathbf{1}))$$

The update equations of the weight vector include the calculation of a matrix inverse of Υ which was originally calculated using the Cholesky decomposition. To keep our objective of small matrices we will instead calculate an SVD based inverse of this matrix using a Nyström approximation of Υ. It should be noted at this point that the matrix Υ is psd by construction. We approximate Υ by selecting another set of m^* landmarks from the indices of the not yet pruned weights and calculate the matrix $\tilde{\Upsilon} = C_{(N,m^*)} W^{-1}_{(m^*,m^*)} C^\top_{(N,m^*)}$ in analogy to Eq (10) with submatrices: [9]

$$\begin{aligned} C_{(N,m^*)} &= E_{(N,[m])} + ((\hat{K_1} \cdot (K_2 \cdot (K_1 \cdot \hat{K}_{2(\cdot,[m^*])}))(\hat{\mathbf{y}}^\top \hat{\mathbf{y}}_{[m^*]})) \\ &\quad \circ\sqrt{2}\hat{\mathbf{w}}) \circ \sqrt{2}\hat{\mathbf{w}}^\top_{[m^*]} \\ W_(m^*, m^*) &= C^{-1}_{(m^*,\cdot)} \end{aligned}$$

[9] The number of landmarks m^* is fixed to be 1% of $|w|$ but not more then 500 landmarks. If the length of $\mathbf{w}$ drops below 100 points we use the original PCVM formulations.

ACCEPTED MANUSCRIPT

Where $\circ$ indicates (in analogy to its previous meaning) that each row of the left matrix is elementwise multiplied by the right vector and $E_{(N,[m])}$ is the matrix consisting of the m landmark columns of the $N \times N$ identity matrix. The terms $\sqrt{2}\hat{\mathbf{w}}$ and $\sqrt{2}\hat{\mathbf{w}}^{\top}_{[m^*]}$ are the entries of the diagonal matrix M as defined in Eq. (7) but now given in vector form.

These two matrices serve as the input of a Nyström approximation based inverse (as discussed in sub section 3.1) and we obtain matrices $V \in \mathbb{R}^{N\times r}, U \in \mathbb{R}^{r\times N}$ and $S \in \mathbb{R}^{r\times r}$, where $r \leq m^*$ is the rank of the inverse. Further we define two vectors

$$\begin{aligned} \mathbf{v}_1 &= \bar{\hat{\mathbf{H}}}_{\theta}^{\top} \cdot K_1 \\ \mathbf{v}_2 &= \mathbf{1}^{\top} \cdot K_1. \end{aligned}$$

We obtain the approximated weight update

$$\mathbf{w}^{\text{new}} = V \cdot (S \cdot U^{\top} \cdot (\sqrt{2}\hat{\mathbf{w}}(\hat{\mathbf{y}}(\mathbf{v}_1 \cdot \hat{K_2})^{\top} - b \cdot \hat{\mathbf{y}}(\mathbf{v}_2 \cdot \hat{K_2})^{\top})))\sqrt{2}\hat{\mathbf{w}}$$

The original bias update (6) is replaced with:

$$\mathbf{b} = t(1+tNt)^{-1}t(\mathbf{1}^{\top}\bar{\hat{\mathbf{H}}}_{\theta} - \mathbf{1}^{\top}((((\hat{\mathbf{y}} \circ \hat{\mathbf{w}})^{\top}\hat{K_1}) \cdot K_2)^{\top}))$$

Subsequently the entries in $\hat{\mathbf{w}}$ which are close to zero are pruned out and the matrices $\hat{K_1}$ and $\hat{K_2}$ are modified accordingly.

5.2. Nyström based Indefinite Kernel Fisher Discriminant

Given a Nyström approximated kernel matrix a few adaptations have to be made to obtain a valid iKFD formulation solely based on the Nyström approximated kernel, without any full matrix operations.

First we need to calculate the classwise means μ_+ and μ_- based on the row/column sums of the approximated input kernel matrix. This can be done by rather simple matrix operations on the two low rank matrices of the Nyström approximation of K. For ease of presentation, we will refer to the matrices $K_{(N,m)}$ and $K_{(m,m)}$ as Ψ and Γ, respectively. Then

$$\sum_i \tilde{K}_{k,i} = \sum_{l=1}^{m}\left(\sum_{j=1}^{N}\Psi_{j,\cdot}\Gamma^{-1}\right)\Psi_{l,k}^{\top}. \tag{14}$$

ACCEPTED MANUSCRIPT

This can obviously also be done in a single matrix operation for all rows in a batch, with linear complexity only. Based on these mean estimates we can calculate Eq. (2). In the next step we need to calculate a squared approximated kernel matrix for the positive and the negative classes, centered at the origin (i.e. with substracted means μ_+ or μ_-). For the positive class with n_+ entries, we can define a new Nyström approximated (squared) matrix with subtracted mean as :

$$\hat{K}^+_{(N,m)} = K_{(N,m)} \cdot K^{-1}_{(m,m)} \cdot (K^\top_{(I_+,m)} \cdot K_{(I_+,m)}) \cdot K^{-1}_{(m,m)} \cdot K^\top_{(m,m)} - \mu_+ \cdot \mu_+^\top \cdot n_+ \quad (15)$$

An equivalent term can be derived for the negative class providing $\hat{K}^-_{(N,m)}$. It should be noted that no obtained matrix in Eq (15) has more than $N \times m$ entries. Finally $\hat{K}^+_{(N,m)}$ and $\hat{K}^-_{(N,m)}$ are combined to approximate the within class matrix as shown in Eq. (3). From the derivation in [4] we know, that only the eigenvector of the Nyström approximated kernel matrix based on $\hat{K}_{(N,m)} = \hat{K}^+_{(N,m)} + \hat{K}^-_{(N,m)}$ are needed. Using a Nyström based eigen-decomposition (explained before) on $\hat{K}_{(N,m)}$ we obtain:

$$\alpha = C \cdot A^{-1} \cdot (C' \cdot (\mu_+ - \mu_-))$$

where C contains the eigenvectors and A the eigenvalues of $\hat{K}_{(N,m)}$. If A is not regular, instead of A^{-1} one can use a pseudo inverse. The bias term b is obtained as $b = -\alpha^\top(\mu_+ + \mu_-)/2$.

6. Complexity analysis

The original iKFD update rules have costs of $\mathcal{O}(N^3)$ and memory storage $\mathcal{O}(N^2)$, where N is the number of points. The Ny-iKFD may involve the extra Nyström approximation of the kernel matrix to obtain $K_{(N,m)}$ and $K^{-1}_{(m,m)}$, if not already given. If we have m landmarks, $m \ll N$, this gives costs of $\mathcal{O}(mN)$ for the first matrix and $\mathcal{O}(m^3)$ for the second, due to the matrix inversion. Further both matrices are multiplied within the optimization so we get $\mathcal{O}(m^2N)$. Similarly, the matrix inversion of the original iKFD with $\mathcal{O}(N^3)$ is reduced to $\mathcal{O}(m^2N) + \mathcal{O}(m^3)$ due to the Nyström approximation of the inverse. If we assume $m \ll N$ the overall runtime and memory complexity of Ny-iKFD is linear in N. For the Ny-PCVM we obtain a similar analysis as shown in [11] but with extra costs to calculate the Nyström approximated SVD.

ACCEPTED MANUSCRIPT

Additionally, Ny-PCVM uses an iterative optimization scheme to optimize and sparsify w with constant costs C_I, as the number of iterations. Accordingly Ny-iKFD and Ny-PCVM have both linear memory and runtime complexity $\mathcal{O}(N)$, but Ny-PCVM maybe slower than Ny-iKFD due to extra overhead costs. The MEB approximation has a linear (worst case) complexity [30] which in our case scales with the constant number of classes C, hence the complexity remains linear.

7. Experiments

We compare iKFD, Ny-iKFD, Ny-PCVM and PCVM on various larger indefinite proximity data. In contrast to many standard kernel approaches, for iKFD and PCVM, the indefinite kernel matrices need not to be corrected by costly eigenvalue correction [34, 35] [10]

Further the iKFD and PCVM provides direct access to probabilistic classification decisions. First we show a small simulated experiment for two Gaussians which exist in an intrinsically two dimensional *pseudo*-Euclidean space $\mathbb{R}^{(1,1)}$. The plot in Figure 7 shows a typical result for the obtained decision planes using the iKFD or Ny-iKFD. The Gaussians are slightly overlapping and both approaches achieve a good separation with 93.50% and 88.50% prediction accuracy, respectively.

Subsequently we consider a few public available datasets for some real life experiments. The data are *Gesture* (1500pts, 20 classes), *Zongker* (2000pts, 10 classes) and *Proteom* (2604pts, 53 classes (restricted to classes with at least 10 entries)) from [36]; *Chromo* (4200pt, 21 classes) from [37] and the SwissProt database *Swiss* (10988 pts, 30 classes) from [38], (version 10/2010, reduced to prosite labeled classes with at least 100 entries). Further we used the *Sonatas* data (1068pts, 5 classes) taken from [39]. All data are processed as indefinite kernels and the landmarks are selected using the respective landmark selection schemes. The mean number of Nyström landmarks as obtained by the MEB approach is given in brackets after the dataset label. For all experiments we report mean and standard errors as obtained by a 10 fold crossvalidation.

[10] In [10] various correction methods have been studied on the same data indicating that eigenvalue corrections may be helpful.

ACCEPTED MANUSCRIPT

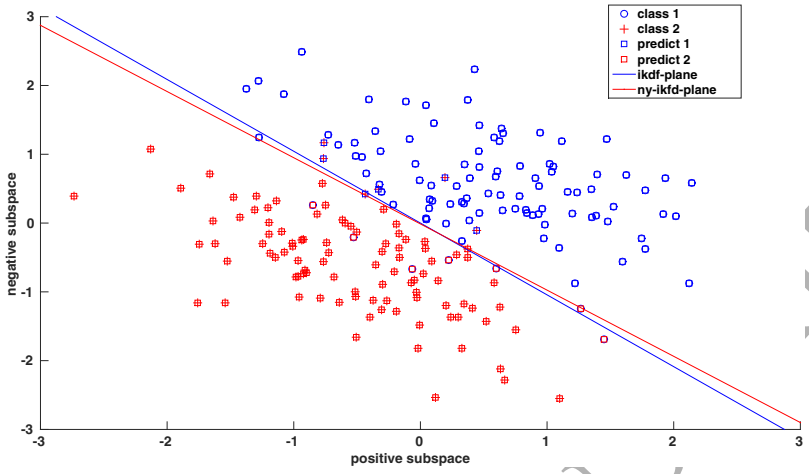


Figure 7: Visualization of the indefinite Fisher kernel for two Gaussians in a two dimensional pseudo-Euclidean space $\mathbb{R}^{(1,1)}$. The predicted labels are with respect to the iKFD classification.

For PCVM we fixed the upper number of optimization cycles to 500. The probabilistic outputs can be directly used to allow for a reject region but can also be used to provide alternative classification decisions e.g. in a ranking framework

In Table 3, 4 and Table 5 we show the results for different non-metric proximity datasets using Ny-PCVM, PCVM and iKFD or Ny-iKFD. The overall best results for a dataset are underlined and the best approximations are highlighted in bold.

Considering Table 3 and Table 4 we see that iKFD and PCVM are similarly effective with slightly better results for iKFD. The Nyström approximation of the kernel matrix *only*, often leads to a in general small decrease of the accuracy, but the additional approximation step, in the algorithm itself, does not substantially decrease the prediction accuracy further[11].

The approximations used in the algorithms Ny-iKFD and Ny-PCVM appear to be effective. The runtime analysis in Table 5 clearly shows that the classical iKFD is very complex. As expected, the integration of the Nyström approximation leads to substantial speed-ups. Larger datasets like the Swiss data with ≈ 10.000 entries could

[11] Also the runtime and model complexity are similar and therefore not reported in the following.

ACCEPTED MANUSCRIPT

dataset	iKFD	(MEB)	(KM)	(ENT)
gesture $1500 \rightarrow 64$	$\underline{97.93} \pm 0.73$	$\mathbf{96.60 \pm 1.84}$	95.73 ± 0.86	93.47 ± 1.93*
sonatas $1068 \rightarrow 25$	90.17 ± 2.14	$83.52 \pm 2.08*$	$77.63 \pm 3.19*$	80.24 ± 2.46*
zongker $2000 \rightarrow 41$	$\underline{96.60} \pm 1.97$	$\mathbf{90.70 \pm 2.30}*$	$88.40 \pm 1.33*$	90.90 ± 1.15*
proteom $2604 \rightarrow 123$	99.58 ± 0.38	$\underline{\mathbf{99.68}} \pm \mathbf{0.31}$	94.78 ± 1.89	94.54 ± 1.87
chromo $4200 \rightarrow 65$	$\underline{97.24} \pm 0.94$	$\mathbf{94.79 \pm 1.45}$	94.17 ± 0.86	94.50 ± 1.30
swiss $10988 \rightarrow 116$	–	$\underline{83.05} \pm 1.60$	73.74 ± 0.71	

Table 3: Comparison of the test set accuracy of iKFD with different input kernels. The first column (iKFD) refers to the results obtained by a full, unapproximated kernel with classical iKFD. The other columns report results for the Ny-iKFD approach with differently approximated input kernels. (MEB) gives results for the proposed approach, (KM) shows results of the kmeans strategy and (ENT) employs the entropy approach. Below the dataset label we provide the number of samples and the number of landmarks used to represent the kernel with MEB, KM and ENT. (*) indicate significant differences with respect to the same unapproximated method. Best approximation results are in bold. Best overall results are underlined. Bold markings indicate the best approximated solution.

not be analyzed by iKFD or PCVM before. We also see that the landmark selection scheme using MEB is slightly more effective than by using k-means but without the need to tune the number of clusters (landmarks). The entropy approach is similar efficient than the k-means strategy but more costly due to the iterative optimization of the landmark set and the respective eigen-decompositions (see [16]).

The PCVM is focusing on a sparse parameter vector $\mathbf{w}$ in contrast to the iKFD. For the iKFD most training points are also used in the model ($\geq 94\%$) whereas for Ny-PCVM often less than 5% are kept in general as shown in Table 6. In practice it is often costly to calculate the non-metric proximity measures like sequence alignments and also a large number of kernel expansions should be avoided. Accordingly sparse

ACCEPTED MANUSCRIPT

dataset	PCVM	(MEB)	(KM)	(ENT)
gesture $1500 \rightarrow 64$	73.20 ± 18.12	$85.53 \pm 1.22*$	$92.60 \pm 1.04*$	91.07 ± 2.97*
sonatas $1068 \rightarrow 25$	91.20 ± 2.69	$\mathbf{87.08 \pm 3.19}*$	$77.81 \pm 3.28*$	82.77 ± 2.86*
zongker $2000 \rightarrow 41$	93.60 ± 2.00	$84.35 \pm 2.53*$	$88.30 \pm 2.89*$	90.50 ± 2.12
proteom $2604 \rightarrow 123$	99.58 ± 0.38	99.45 ± 0.53	94.18 ± 1.23	80.93 ± 22.96*
chromo $4200 \rightarrow 65$	93.29 ± 1.51	92.21 ± 1.31	92.10 ± 0.89	90.95 ± 2.55
swiss $10988 \rightarrow 116$	–	70.38 ± 19.19	75.36 ± 7.55	

Table 4: Comparison of the test set accuracy of PCVM with different input kernels. The first column (PCVM) refers to the results obtained by a full, unapproximated kernel with classical PCVM. The other columns report results for the Ny-PCVM approach with differently approximated input kernels. (MEB) gives results for the proposed approach, (KM) shows results of the kmeans strategy and (ENT) employs the entropy approach. Below the dataset label we provide the number of samples and the number of landmarks used to represent the kernel with MEB, KM and ENT. (*) indicate significant differences with respect to the same unapproximated method. Best approximation results are in bold. Best overall results are underlined. Bold markings indicate the best approximated solution.

models are very desirable. Considering the runtime again Ny-PCVM and Ny-iKFD are in general faster than the original algorithms, typically by at least a magnitude. the PCVM and Ny-PCVM are also very fast in the test case or out-of sample extension due to the inherent model sparsity.

In [9] and [10] one can also find an in depth analysis of alternative non-probabilistic classifiers and how they perform on the considered data sets. Overall the accuracy of our approaches is competitive to other reported results. These alternative techniques have in general quadratic to cubic complexity, are often non-sparse in the final model and are more complicated to handle if the model is applied to new test data. In particular the work in [9] provides a large disussion about the practical issues of handling

ACCEPTED MANUSCRIPT

	iKFD	Ny-iKFD	PCVM	Ny-PCVM
gesture	50.72 ± 1.54	9.18 ± 0.19	116.33 ± 7.49	31.98 ± 0.42
sonatas	5.04 ± 0.22	1.85 ± 0.06	60.07 ± 2.54	7.01 ± 0.24
zongker	51.61 ± 1.43	5.53 ± 0.16	184.07 ± 14.97	16.91 ± 0.24
proteom	559.25 ± 15.29	42.08 ± 1.92	352.08 ± 18.05	111.22 ± 1.88
chromo	763.24 ± 31.54	27.91 ± 1.77	694.43 ± 15.61	54.36 ± 0.77
swiss	–	178.79 ± 10.63	–	123.29 ± 2.72

Table 5: Typical runtimes (in sec.) - indefinite kernels

	iKFD	Ny-iKFD (MEB)	PCVM	Ny-PCVM (MEB)
gesture	100.00 ± 0	100.00 ± 0	10.60 ± 0.84	5.25 ± 0.31
sonatas	100.00 ± 0	100.00 ± 0	11.24 ± 0.56	3.42 ± 0.57
zongker	100.00 ± 0	100.00 ± 0	14.42 ± 3.65	8.63 ± 0.31
proteom	100.00 ± 0	100.00 ± 0	5.23 ± 0.36	5.85 ± 0.14
chromo	100.00 ± 0	100.00 ± 0	7.49 ± 0.51	2.49 ± 0.34
swiss	–	96.95 ± 0.27	–	1.18 ± 0.25

Table 6: Model complexity - indefinite kernels (threshold $1e^{-4}$)

ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

non-psd kernels with the Support Vector Machine and was a motivation for our work.

8. Conclusions

We presented an alternative formulation of the iKFD and PCVM employing the Nyström approximation. We also provided an alternative way to identify the landmark points of the Nyström approximation in cases where the objective is a supervised problem. Our results indicate that in general the MEB approach is similar efficient compared to the k-means clustering or the entropy strategy but with less effort and almost parameter free. We found that Ny-iKFD is competitive in the prediction accuracy with the original iKFD and alternative approaches, while taking substantially less memory and runtime but being less sparse then Ny-PCVM. The Ny-iKFD and Ny-PCVM provides now an effective way to obtain a *probabilistic* classification model for medium to large psd *and* non-psd datasets, in *batch mode* with *linear* runtime and memory complexity. If sparsity is not an issue one may prefer Ny-iKFD which is slightly better in the prediction accuracy then Ny-PCVM. Using the presented approach we believe that iKFD is now applicable for realistic problems and may get a larger impact then before. In future work it could be interesting to incorporate sparsity concepts into iKFD and Ny-iKFD similar as shown for classical KFD in [40].

Implementation: The Nyström approximation for iKFD is provided at `http://www.techfak.uni-bielefeld.de/~fschleif/source/ny_ikfd.tgz` and the PCVM/Ny-PCVM code can be found at `https://mloss.org/software/view/610/`.

Acknowledgment: A Marie Curie Intra-European Fellowship (IEF): FP7-PEOPLE-2012-IEF (FP7-327791-ProMoS) and support from the Cluster of Excellence 277 Cognitive Interaction Technology funded by the German Excellence Initiative is gratefully acknowledged. PT was supported by the EPSRC grant EP/L000296/1, "Personalized Health Care through Learning in the Model Space". We would like

ACCEPTED MANUSCRIPT

to thank R. Duin, Delft University for various support with distools and prtools and Huanhuan Chen,University of Science and Technology of China, for providing support with the Probabilistic Classification Vector Machine.

References

[1] T. F. Smith, M.S., Waterman, Identification of common molecular subsequences., Journal of molecular biology 147 (1) (1981) 195–197.

[2] M.-P. Dubuisson, A. Jain, A modified hausdorff distance for object matching, in: Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision amp; Image Processing., Proceedings of the 12th IAPR International Conference on, Vol. 1, 1994, pp. 566–568 vol.1.

[3] H. Ling, D. W. Jacobs, Shape classification using the inner-distance, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2) (2007) 286–299. doi:10.1109/TPAMI.2007.41.
URL http://doi.ieeecomputersociety.org/10.1109/TPAMI.2007.41

[4] B. Haasdonk, E. Pekalska, Indefinite kernel fisher discriminant, in: 19th International Conference on Pattern Recognition (ICPR 2008), December 8-11, 2008, Tampa, Florida, USA, IEEE Computer Society, 2008, pp. 1–4. doi:10.1109/ICPR.2008.4761718.
URL http://dx.doi.org/10.1109/ICPR.2008.4761718

[5] E. Pekalska, B. Haasdonk, Kernel discriminant analysis for positive definite and indefinite kernels, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (6) (2009) 1017–1031.
URL http://www.scopus.com/inward/record.url?eid=2-s2.0-65549157171&partnerID=40&md5=1dfbac0ec84c42175c3c9ba88976fb76

[6] H. Chen, P. Tino, X. Yao, Probabilistic classification vector machines, IEEE Transactions on Neural Networks 20 (6) (2009) 901–914.

ACCEPTED MANUSCRIPT

[7] B. Haasdonk, Feature space interpretation of svms with indefinite kernels, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (4) (2005) 482–492.
URL http://www.scopus.com/inward/record.url?eid=2-s2.0-17144429687&partnerID=40&md5=e0fcfe5f309a9e8236d3ff5a8cf2d920

[8] I. M. Alabdulmohsin, X. Gao, X. Zhang, Support vector machines with indefinite kernels, in: D. Q. Phung, H. Li (Eds.), Proceedings of the Sixth Asian Conference on Machine Learning, ACML 2014, Nha Trang City, Vietnam, November 26-28, 2014., Vol. 39 of JMLR Proceedings, JMLR.org, 2014.
URL http://jmlr.org/proceedings/papers/v39/alabdulmohsin14.html

[9] G. Loosli, S. Canu, C. S. Ong, Learning svm in krein spaces, IEEE Transactions on Pattern Analysis and Machine Intelligence PP (99) (2015) 1–1. doi:10.1109/TPAMI.2015.2477830.

[10] F.-M. Schleif, P. Tino, Indefinite proximity learning - a review, Neural Computation 27 (10) (2015) 2039–2096.

[11] P. T. F.-M. Schleif, A.Gisbrecht, Probabilistic classification vector machine at large scale, in: Proceedings of ESANN 2015, 2015, p. to appear.

[12] A. Gisbrecht, F.-M. Schleif, Metric and non-metric proximity transformations at linear costs, Neurocomputing 167 (2015) 643–657.

[13] C. K. I. Williams, M. Seeger, Using the nyström method to speed up kernel machines, in: NIPS 2000, 2000, pp. 682–688.

[14] K. Zhang, J. T. Kwok, Clustered nyström method for large scale manifold learning and dimension reduction, IEEE Transactions on Neural Networks 21 (10) (2010) 1576–1587.

[15] S. Si, C. Hsieh, I. S. Dhillon, Memory efficient kernel approximation, in: Proceedings of the 31th International Conference on Machine Learning, ICML 2014,

ACCEPTED MANUSCRIPT

Beijing, China, 21-26 June 2014, Vol. 32 of JMLR Proceedings, JMLR.org, 2014, pp. 701–709.
URL http://jmlr.org/proceedings/papers/v32/si14.html

[16] K. D. Brabanter, J. D. Brabanter, J. A. K. Suykens, B. D. Moor, Optimized fixed-size kernel models for large data sets, Computational Statistics & Data Analysis 54 (6) (2010) 1484–1504. doi:10.1016/j.csda.2010.01.024.
URL http://dx.doi.org/10.1016/j.csda.2010.01.024

[17] P. Drineas, M. Magdon-Ismail, M. W. Mahoney, D. P. Woodruff, Fast approximation of matrix coherence and statistical leverage, Journal of Machine Learning Research 13 (2012) 3475–3506.
URL http://dl.acm.org/citation.cfm?id=2503352

[18] F. Schleif, A. Gisbrecht, P. Tiňo, Large scale indefinite kernel fisher discriminant, in: A. Feragen, M. Pelillo, M. Loog (Eds.), Similarity-Based Pattern Recognition - Third International Workshop, SIMBAD 2015, Copenhagen, Denmark, October 12-14, 2015, Proceedings, Vol. 9370 of Lecture Notes in Computer Science, Springer, 2015, pp. 160–170. doi:10.1007/978-3-319-24261-3_13.
URL http://dx.doi.org/10.1007/978-3-319-24261-3_13

[19] J. Shawe-Taylor, N. Cristianini, Kernel Methods for Pattern Analysis and Discovery, Cambridge University Press, 2004.

[20] B. Haasdonk, D. Keysers, Tangent distance kernels for support vector machines, in: ICPR (2), 2002, pp. 864–868.

[21] A. Cichocki, S.-I. Amari, Families of alpha- beta- and gamma- divergences: Flexible and robust measures of similarities, Entropy 12 (6) (2010) 1532–1568.

[22] E. Pekalska, R. Duin, The dissimilarity representation for pattern recognition, World Scientific, 2005.

[23] L. Goldfarb, A unified approach to pattern recognition, Pattern Recognition 17 (5) (1984) 575 – 582.

ACCEPTED MANUSCRIPT

[24] J. Yang, L. Fan, A novel indefinite kernel dimensionality reduction algorithm: Weighted generalized indefinite kernel discriminant analysis, Neural Processing Letters (2013) 1–13.
URL http://www.scopus.com/inward/record.url?eid=2-s2.0-84887547713&partnerID=40&md5=e80956c105c9523239fe251ef25669b6

[25] H. Chen, P. Tino, X. Yao, Efficient probabilistic classification vector machine with incremental basis function selection, IEEE TNN-LS 25 (2) (2014) 356–369.

[26] K. Zhang, I. W. Tsang, J. T. Kwok, Improved Nystrom low-rank approximation and error analysis, in: Proceedings of the 25th international conference on Machine learning, ICML '08, ACM, New York, NY, USA, 2008, pp. 1232–1239.

[27] A. Gittens, M. W. Mahoney, Revisiting the nyström method for improved large-scale machine learning, CoRR abs/1303.1849.

[28] J. T. Kwok, I. W. Tsang, Learning with idealized kernels, in: T. Fawcett, N. Mishra (Eds.), Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA, AAAI Press, 2003, pp. 400–407.
URL http://www.aaai.org/Library/ICML/2003/icml03-054.php

[29] I. W. Tsang, J. T. Kwok, P. Cheung, Core vector machines: Fast SVM training on very large data sets, Journal of Machine Learning Research 6 (2005) 363–392.
URL http://www.jmlr.org/papers/v6/tsang05a.html

[30] M. Badoiu, K. L. Clarkson, Optimal core-sets for balls, Comput. Geom. 40 (1) (2008) 14–22.

[31] R. P. W. Duin, E. Pekalska, Non-euclidean dissimilarities: Causes and informativeness, in: Structural, Syntactic, and Statistical Pattern Recognition, Joint IAPR International Workshop, SSPR&SPR 2010, Cesme, Izmir, Turkey, August 18-20, 2010. Proceedings, 2010, pp. 324–333.

ACCEPTED MANUSCRIPT

[32] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Computation 15 (6) (2003) 1373–1396. doi:10.1162/089976603321780317.
URL http://dx.doi.org/10.1162/089976603321780317

[33] T. Villmann, J. C. Claussen, Magnification control in self-organizing maps and neural gas, Neural Computation 18 (2) (2006) 446–469. doi:10.1162/089976606775093918.
URL http://dx.doi.org/10.1162/089976606775093918

[34] Y. Chen, E. K. Garcia, M. R. Gupta, A. Rahimi, L. Cazzanti, Similarity-based classification: Concepts and algorithms, JMLR 10 (2009) 747–776.

[35] F.-M. Schleif, A. Gisbrecht, Data analysis of (non-)metric proximities at linear costs, in: Proceedings of SIMBAD 2013, 2013, pp. 59–74.
URL pdf/simbad_2013.pdf

[36] R. P. Duin, PRTools (march 2012).
URL http://www.prtools.org

[37] M. Neuhaus, H. Bunke, Edit distance based kernel functions for structural pattern classification, Pattern Recognition 39 (10) (2006) 1852–1863.

[38] B. Boeckmann, A. Bairoch, R. Apweiler, M.-C. Blatter, A. Estreicher, E. Gasteiger, M. Martin, K. Michoud, C. O'Donovan, I. Phan, S. Pilbout, M. Schneider, The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003,, Nucleic Acids Research 31 365–370.

[39] B. Mokbel, A. Hasenfuss, B. Hammer, Graph-based representation of symbolic musical data, in: A. Torsello, F. Escolano, L. Brun (Eds.), Graph-Based Representations in Pattern Recognition, 7th IAPR-TC-15 International Workshop, GbRPR 2009, Venice, Italy, May 26-28, 2009. Proceedings, Vol. 5534 of Lecture Notes in Computer Science, Springer, 2009, pp. 42–51. doi:10.1007/978-3-642-02124-4_5.
URL http://dx.doi.org/10.1007/978-3-642-02124-4_5

ACCEPTED MANUSCRIPT

[40] T. Diethe, Z. Hussain, D. R. Hardoon, J. Shawe-Taylor, Matching pursuit kernel fisher discriminant analysis, in: D. A. V. Dyk, M. Welling (Eds.), Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics, AISTATS 2009, Clearwater Beach, Florida, USA, April 16-18, 2009, Vol. 5 of JMLR Proceedings, JMLR.org, 2009, pp. 121–128.
URL http://www.jmlr.org/proceedings/papers/v5/diethe09a.html

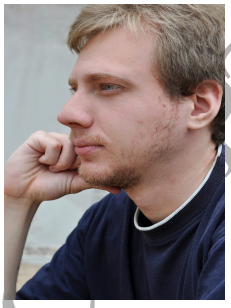
ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

Author Biographies

803

Frank-Michael Schleif (Dipl.-Inf, University of Leipzig, PhD, TU-Clausthal, Germany) was a Marie Curie Senior Research Fellow at the University of Birmingham, Birmingham, UK and a Post-Doctoral Fellow in the group of Theoretical Computer Science (TCS) at the University of Bielefeld, Bielefeld, Germany, where he also received a venia legendi in applied computer science in 2013. He was also a software developer and consultant for the Bruker Corp. Since 2016 he is with the University of Applied Sciences, Wuerzburg, Germany, where he is a Professor for Database Management and Business Intelligence. His current research interests include data management, computational intelligence techniques and machine learning for non-metric models and large scale problems. Several research stays have taken him to UK, the Netherlands, Japan and the USA. He is a member of the German chapter of the European Neural Network Society (GNNS), the GI and the IEEE-CIS. He is editor of the Machine Learning Reports and member of the editorial board of the Neural Processing Letters.



Andrej Gisbrecht (Dipl.-Inf., Clausthal University of Technology; PhD. with distinction from the Cognitive Interaction Technology Center of Excellence at Bielefeld University, Germany) is currently a postdoc at the Probabilistic Machine Learning Group at Aalto University, Finland. Several research stays took him to the Aalto University in Finland, to the University of Groningen, NL and to the University of Birmingham in UK. He has been invited to several research seminars at Dagstuhl, MPI in Dresden and university groups. With his research he contributed to the areas of visualisation, big data and time series analysis.

ACCEPTED MANUSCRIPT

Peter Tino (M.Sc. Slovak University of Technology, Ph. D. Slovak Academy of Sciences) was a Fulbright Fellow with the NEC Research Institute, Princeton, NJ, USA, and a Post-Doctoral Fellow with the Austrian Research Institute for AI, Vienna, Austria, and with Aston University, Birmingham, UK. Since 2003, he has been with the School of Computer Science, University of Birmingham, Edgbaston, Birmingham, UK, where he is currently a Full Professor-Chair in Complex and Adaptive Systems. His current research interests include dynamical systems, machine learning, probabilistic modelling of structured data, evolutionary computation, and fractal analysis. Peter was a recipient of the Fulbright Fellowship in 1994, the U.K.-Hong-Kong Fellowship for Excellence in 2008, three Outstanding Paper of the Year Awards from the IEEE Transactions on Neural Networks in 1998 and 2011 and the IEEE Transactions on Evolutionary Computation in 2010, and the Best Paper Award at ICANN 2002. He serves on the editorial boards of several journals.

ACCEPTED MANUSCRIPT